Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 52 (2015) 381 – 388

Procedia Computer Science

6th International Conference on Ambient Systems, Networks and Technologies (ANT 2015)

Fast Emulation of Self-Organizing Maps for Large Datasets

Macario O. Cordel II*, Arnulfo P. Azcarraga

De La Salle University, 2401 Taft Avenue, Manila, 1004, Philippines

Abstract

The self-organizing map (SOM) methodology does vector quantization and clustering on the dataset, and then projects the obtained clusters to a lower dimensional space, such as a 2D map, by positioning similar clusters in locations that are spatially closer in the lower dimension space. This makes the SOM methodology an effective tool for data visualization. However, in a world where mined information from big data have to be available immediately, SOM becomes an unattractive tool because of its time complexity. In this paper, we propose an alternative visualization methodology for large datasets that emulates SOM methodology without the speed constraints inherent to SOM. To demonstrate the efficiency and the potential of the proposed scheme as a fast visualization tool, the methodology is used to cluster and project the 3,823 image samples of handwritten digits of the Optical Recognition of Handwritten Digits dataset. Although the dataset is not, by any means large, it is sufficient to demonstrate the speed-up that can be achieved by using this proposed SOM emulation procedure.

Peer-review under responsibility of the Conference Program Chairs

Keywords: Data visualization; self-organizing map; multidimensional scaling; two-level clustering; fast data analysis; positions of clusters;

1. Introduction

One of the enablers of Big data is the intuitive presentation of information such as visualization [1]. Visualization provides intuitive display of unstructured information e.g. emails, text messages, audio as well as video streams. These types of unstructured data continuously grow requiring visualization tools to have more efficient running performance. One of these visualization tools is the Self-Organizing Map (SOM) [2].

SOM represents data using nodes as points in the two-dimensional (or three-dimensional) vector space. These SOM nodes have weight vectors which are updated per iteration depending on the input vector from the data set. Generally, the weight vectors are updated as follows.

$$\mathbf{w}_i(t+1) = \mathbf{w}_i(t) + G(t)\alpha_i(t)||\mathbf{x}(t) - \mathbf{w}_i(t)|| \quad (1)$$

where t represents the iteration number, $\mathbf{w}_i$ represents the weight vector of the ith node, $\mathbf{x}(t)$ is the input vector chosen randomly from the training set, $\alpha_i(t)$ is the learning rate of the adaptation process, $G(t)$ is a window function which

* Corresponding author. Tel.: +632-524-4611 loc. 306 ; fax: +632-526-4247.
E-mail address: macario.cordel@dlsu.edu.ph

Peer-review under responsibility of the Conference Program Chairs
doi:10.1016/j.procs.2015.05.002

is typically a Gaussian window or a rectangular window, and $\|\mathbf{x}(t) - \mathbf{w}_i(t)\|$ is the Euclidean distance between $\mathbf{x}(t)$ and $\mathbf{w}_i(t)$. The intuitive display of the datas relative distance, distribution and clusters make SOM an attractive tool for data visualization. However, for large dataset, Equation 1 has to be performed several times, increasing SOMs complexity.

For N^* number of SOM nodes with M weights per node and N number of samples, the computational requirement is $\mathbf{O}(N^2 \times N^* \times M)$ for distance computation, $\mathbf{O}(N \times N^* \log N^*)$ for winning node selection, and $\mathbf{O}(N^2 \times N^* \times M)$ for weight update computation using a Gaussian window.

An alternative and simpler data visualization tool, called the multidimensional scaling (MDS), makes use of singular-value decomposition (SVD) for data mapping to remove the need for iteration which is highly based on the number of samples. The MDS reveals the structure of a data set, typically high dimensional data, by transforming the pairwise dissimilarities of each element (in the dataset) into distances in low dimensional vector space[3,4,5]. Recent works[6,7,8,9] on wireless sensor nodes (WSN) make use of MDS on node localization problem where only the nodes receive signal information are known. Despite its applicability to complex problems, e.g. in marketing[10] and wireless networks[11] MDS requires N^2 amount of memory and distance computations, which make it impractical to use for large datasets. Furthermore, it lacks clustering and distribution information which make it ineffective data visualization tool.

Thus, we present in this work an alternative data visualization methodology to overcome the time complexity issue of the SOM in large number of samples and the limited information provided by the MDS as a projection tool. This proposed scheme is discussed in section 2. To demonstrate its vast potential as a visualization tool, an experiment is performed using the Optical Recognition of Handwritten Digits dataset[12]. Results and analysis of which are presented in section 3 followed by the conclusion and future works in section 4.

Nomenclature

M	The number of attributes per SOM node / dimension of the attribute vector of the original dataset
N	Number of samples in the large dataset
N^*	Number of SOM nodes / number of prototypes in the proposed approach equal to K
K	Number of prototypes, equal to N^*, associated with k-means in phase 1
k	Number of clusters of prototypes in phase 2, associated with k-means in phase 2
$^{(\bullet)}$	Superscript $(\bullet)$ denotes the dimension of the vector

2. Large data visualization methodology

Consider a large database of M-dimensional data with N samples whose attribute vector is denoted by $\phi_i^{(M)}$, where $i = 1, 2, \ldots, N$. The relative Euclidean distance measurement between two data entries i and j of the given data set is given by

$$D = [d_{ij}] = \|\phi_i^{(M)} - \phi_j^{(M)}\| \tag{2}$$

where $\|\cdot\|$ denotes the Frobenius norm. Applying classical MDS for large value of N requires $N \times N$ memories, e.g. 10^{10} for $N = 10^5$. Applying SOM similarly is impractical. The task is to provide mapping of N high-dimensional data in $R^{(M)}$ onto low-dimensional vector space, e.g. $R^{(2)}$ while providing the clustering and data proximity information.

The proposed scheme is designed to emulate SOM by providing data proximity and clustering information. It is mainly divided into three phases: (1) data summarization into prototypes, (2) clustering of prototypes and (3) data mapping, as shown in Figure 1. The first phase aims to decrease the number of data samples, N, into smaller number of prototypes, N^*, by performing k-means on the large dataset. Since N^* equals the number of prototypes, then N^* equals the number of clusters in this application of k-means. The second phase performs prototype clustering to introduce this information in data mapping. For supervised learning, the number of clusters, called the small k, is usually set to be equal to the number of actual classes in the data. To distinguish k of phase 1 k-means from k of phase 2 k-means, the former is called big K (which is equal to N^*) while the latter is called small k. Finally, phase 3 performs